Diseño y desarrollo de una guía para la implementación de un ambiente Big data en la Universidad Católica de Colombia

Fabián Andrés Guerrero López,faguerrero60@ucatolica.edu.co
Jorge Eduardo Rodríguez Pinilla,jerodriguez84@ucatolica.edu.co.
Programa de Ingeniería de sistemas, Facultad de Ingeniería, Universidad Católica de Colombia
Bogotá D.C., Colombia

Resumen-En la actualidad la información se ha vuelto el recurso más importante en las empresas debido a su capacidad de brindar estrategias y alternativas de negocio. A través del tiempo esta información ha sido manejada según el modelo relacional de bases de datos, el cual ha servido para almacenar datos estructurados, pero con la aparición de nuevas tecnologías en el mercado como las redes sociales y GPS, las cuales implementan datos no estructurados como archivos de audio, imagen y video, surge una problemática respecto al incremento de datos del tipo abstracto. Por tal motivo se hace necesaria la búsqueda de una alternativa para dar solución a las nuevas necesidades concernientes al almacenamiento y administración de información; esta solución es la implementación de Big data como un nuevo enfoque para optimizar los procesos en las organizaciones.

Este artículo tiene como propósito mostrar a través de un ejemplo práctico, como sería una base de datos RDBMS migrada hacia una plataforma NoSQL la cual está orientada a columnas. Teniendo como referencia una base de datos en MySQL la cual almacena la información bancaria de un usuario, de esta manera se busca dar una perspectiva de Big data como una tecnología innovadora y como una alternativa a los modelos que se utilizan actualmente.

Abstract-*Nowadays the information has become the most important resource in companies due to its ability to provide business strategies. Along time the information has been handled by the relational database model, which has been used to store structured data, but with the emergence of new technologies in the market such as social networking and GPS that implement unstructured data as audio and video files, an issue arises regarding the increase in the generation of abstract data type, which represent a new perspective on data interpretation. For this reason it is necessary to search for an alternative to solve the new requirements for the storage and management of information; this solution is the implementation of big data as a new approach to optimize processes in organizations.*

This paper has the purpose to show through a practical example, how a RDBMS database would be, if it is migrated to a oriented column NoSQL platform, using a MySQL Database that stores the information from a bank user, thus seeks to provide a perspective of big data as an innovative technology and an alternative to currently used models.

1. INTRODUCCIÓN

El constante crecimiento de información en cada aspecto como sociedad, comercio y ciencia, demanda una alternativa con respecto a la gestión de datos a través de bases de datos relacionales, ya que este modelo se limita a ambientes donde los datos que se registran son estructurados y de poco volumen, teniendo como consecuencia un incremento en los costos que conlleva analizar esta información. Adicionalmente con la aparición de los tipos de dato no estructurados que encontramos en las redes sociales y dispositivos de geo localización, tales como archivos de video y de audio, conlleva a pensar en que se debe implementar herramientas que permitan administrar y gestionar este tipo de datos, obteniendo beneficios como la identificación de patrones recurrentes para la toma de decisiones.

Este articulo busca brindar una perspectiva sobre Big data, los beneficios que nos puede brindar, las tecnologías que hay disponibles, todo esto enfocado a las bases de datos orientadas a columnas, tomando en cuenta aspectos

relevantes como el software y el hardware que se debe utilizar, buscando así el propósito de diseñar estrategias y patrones que permitan analizar la información de una manera más detallada.

La metodología de investigación utilizada para el desarrollo del artículo es la sintética, puesto que a partir de varios elementos por separado se pretende llegar a un resultado concreto.

La parte inicial del articulo está constituida por los antecedentes en la utilización de Big data en Colombia, seguido de la justificación de la realización del proyecto y por último el desarrollo donde se abarcan temas como: la definición de Big data, conceptos de bases de datos NoSQL, y la implementación del caso de estudio utilizado para el desarrollo del artículo.

2. ANTECEDENTES

Muchas empresas se han dado cuenta de la importancia de utilizar métodos más eficaces y efectivos para administrar su información, buscando realizar un análisis detallado sobre sus productos y servicios con el objetivo de optimizar sus ganancias, por tal motivo la inclusión de Big Data se hace necesaria para estas organizaciones ya que sirve como un respaldo para identificar patrones de comportamiento entre los clientes, así se construyen estrategias para la toma de decisiones basadas en los resultados obtenidos. Este es el caso de empresas colombianas como Activos S.A,. esta empresa implemento una plataforma para soportar sus procesos de negocio, teniendo en cuenta el volumen de información que manejan diariamente. La multinacional Nutresa tiene en su poder un sistema privado de computación alojado en la nube, que le permite unificar sus procesos financieros, logísticos y de marketing entre sus sucursales; Colombina S.A. adquirió una plataforma tecnológica que le permitió mejorar la distribución de información de las ventas en cada una de sus sucursales. Tomando como referencia lo anterior es claro que la implementación de Big Data gana mayor prestigio entre las grandes organizaciones, según la necesidad de procesar y analizar grandes cantidades de información.

3. JUSTIFICACIÓN

La importancia de desarrollar este documento, radica en apoyar el aprendizaje de las nuevas tendencias en la gestión de la información, fomentando de esta forma el campo de investigación dentro de la comunidad universitaria.

4. DESARROLLO DEL PROYECTO

4.1 DEFINICION DE BIG DATA

Big data es una referencia a aquellos sistemas de información que manejan conjuntos de datos de gran volumen, de alta velocidad, de veracidad, de valor y de gran variedad de recursos, que demandan formas rentables e innovadoras de procesamiento de la información para mejorar la comprensión y la toma de decisiones. [1]

Big data es la solución al crecimiento exponencial de los datos, en el momento en que se hace difícil su administración con respecto al almacenamiento, procesamiento y acceso. [2]

De esto se puede obtener beneficios como:
Optimizar el cálculo y la precisión algorítmica para reunir, analizar, enlazar y comparar conjuntos de grandes datos.
Identificar patrones para la toma de decisiones en los ámbitos económico, social, técnico y legal. [3]

La mayoría de las definiciones que se pueden encontrar de Big data están enfocadas al volumen de los datos, al almacenamiento de dicha información, de esto se puede concluir que el volumen importa, pero también existen otros atributos importantes de Big data, estos son la velocidad, la veracidad, la variedad y el valor. Estos cinco aspectos constituyen una definición comprensiva y además destruyen el mito acerca de que Big data se trata únicamente del volumen. A cada uno de estos aspectos se le atribuyen las siguientes características. [4]

Tabla 1. Características de las bases de datos NoSQL

Volumen	Velocidad	Variedad	Veracidad	Valor
Almacenamiento En terabytes	Por lotes	Estructurado	Integridad y Autenticidad	Estadísticas
Registros	Tiempo Cercano	No estructurado	Origen y Reputación	Eventos
Transacciones	Tiempo Real	Multi-factor	Disponibilidad	Correlaciones
Tablas y Archivos	Procesos	Probabilística	Responsabilidad	Hipótesis

4.2 BASES DE DATOS NOSQL

Con la aparición del término NoSQL en los 90's y su primer uso en el 2009 por Eric Vans, se pretende dar una solución a las problemáticas planteadas anteriormente, dando una posibilidad de abordar la forma de gestionar la información de una manera distinta a como se venía realizando. [5]

Para dar una definición adecuada de las bases de datos NoSQL se pude tener en cuenta las siguientes características:

Distribuido: sistemas de bases de datos NoSQL son a menudo distribuidos donde varias máquinas cooperan en grupos para ofrecer a los clientes datos. Cada pieza de los datos se replica normalmente durante varias máquinas para la redundancia y alta disponibilidad.

Escalabilidad horizontal: a menudo se pueden añadir nodos de forma dinámica, sin ningún tiempo de inactividad, lo

que los efectos lineales de almacenamiento logran capacidades de procesamiento general.

Construido para grandes volúmenes: Muchos sistemas NoSQL fueron construidos para ser capaz de almacenar y procesar enormes cantidades de datos de forma rápida.

Modelos de datos no relacionales: Los modelos de datos varían, pero en general, no son relacional. Por lo general, permiten estructuras más complejas y no son tan
Rígida que el modelo relacional.

No hay definiciones de esquema: La estructura de los datos generalmente no se definen a través de esquemas explícitos que la base de datos manejan. En su lugar, los clientes almacenan datos como deseen, sin tener que cumplir con algunas estructuras predefinidas[6]

Dentro de las bases de datos NoSQL se pueden encontrar 4 categorías de acuerdo con la taxonomía propuesta por Scofield y Popescu:

Almacenes Key-Valué: Estas son las bases de datos más simples en cuanto su uso (la implementación puede ser muy complicada), ya que simplemente almacena valores identificados por una clave.

Bases de datos Columnares: estas bases de datos guardan la información en columnas en lugar de filas, con esto se logra una mayor velocidad en realizar la consulta. Solución conveniente en ambientes donde se presenten muchas lecturas como data warehouses.

Bases de datos orientadas a documentos: son un almacén Key-Valué, a diferencia de este la información no se guarda en binario, sino como un formato que la base de datos pueda leer, como XML, permitiendo realizar consultas avanzadas sobre los datos almacenados.

Bases de datos orientados a grafos: estas bases de datos manejan la información en forma de grafo, dando una mayor importancia a la relación que tiene los datos. Con esto se logra que las consultas puedan ser logradas de formas más óptimas que en un modelo relacional [7]

4.3 IMPLEMENTACION CASO DE ESTUDIO

El caso de estudio que se va a implementar dentro del ambiente Big Data, es el modelo de una base de datos relacional sobre el manejo de las transacciones bancarias, al cual se realizara la transformación a una estructura de columnas para adaptarla al motor NoSQL HBase.

Los pasos para la ejecución del caso de estudio son los siguientes:

- Estructura de Bases de datos NoSQL orientadas a columnas
- Transformación del modelo relacional al modelo orientado a columnas

4.3.1 ESTRUCTURA BASES DE DATOS NOSQL ORIENTADA A COLUMNAS

Las bases de datos orientada a columnas son muy similares a las bases de datos relacionales, pero tienen algunas diferencias en la estructura y en el manejo de los datos; por ejemplo en el almacenamiento, las bases de datos relacional almacenan por filas, mientras las orientadas a columnas almacenan los datos a través de columnas, lo cual permite unir información del mismo tipo dentro de una misma tabla para optimizar su consulta.

Los siguientes conceptos son fundamentales para entender cómo funcionan las bases de datos orientadas a columna:

- Familias columna: es la forma de almacenar los datos en el disco. Todos los datos en una sola columna de familia se guardan en el mismo archivo. Una familia de columna puede contener súper-columnas o columnas.
- Súper-columna: es una columna que contiene un conjunto de columnas
- Columna: es una tupla de nombre y valor[8]

A continuación se puede observar un ejemplo para comprender la forma de almacenar los datos en las bases NoSQL orientada a columnas:

Tabla 2. Estructura Bases de datos orientada a columnas

Tabla	Row (clave)	Familia	Súper-columna	Columna
Blog	aa/mm/dd	información	Autor, titulo, subtitulo	Valor de cada columna
		Comentario título:		Descripción del comentario
usuario	Login-nombre	información	Nombre, password	Valor para cada columna

Teniendo en cuenta el cuadro anterior, se observa que dentro de una misma tabla se pueden incluir diferentes tipos de familias (se puede tomar como tablas del modelo relacional), esto permite interpretar la estructura que tiene las bases de datos orientadas a columnas y la manera que se puede adaptar el modelo relacional a través de la unión de tablas.

4.3.2 TRANSFORMACIÓN DEL MODELO RELACIONAL AL MODELO ORIENTADO A COLUMNAS

Para lograr pasar un modelo relacional a un modelo orientado a columnas se debe tener en cuenta dos aspectos importantes: cuales son los datos más primordiales dentro de la base de datos y cuáles son las consultas más

relevantes. Esto permitirá relacionar los datos dentro de una misma familia de columnas, para poder encontrar fácilmente la información.

Es importante resaltar que existen muchas maneras para transformar el mismo modelo relacional, ya que se pueden tomar distintos caminos para la unión de los datos, esto depende de la información que se desee encontrar o saber.
En la siguiente tabla se puede observar los aspectos equivalentes entre el modelo relacional y el modelo orientado a columnas:

Tabla 3. Comparación modelo relacional vs orientado a columnas.

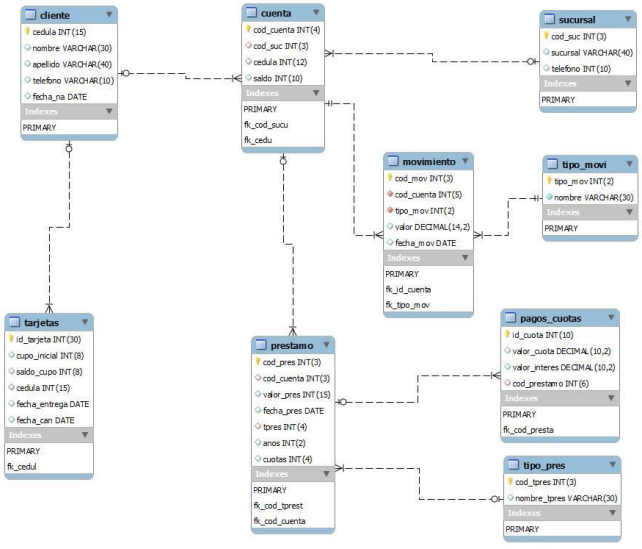
Modelo Relacional	Modelo orientado a columnas
Entidades	Tablas
Atributos	Columnas
Llaves primarias	Row key

Para realizar la trasformación se tomara como ejemplo una base de datos relacional del manejo de las transacciones bancarias, la cual contiene 8 tablas, cada una de ellas con sus respectivos atributos y las relaciones que existen entre las mismas.

En este caso para la trasformación del modelo se tomara como aspecto primordial las transacciones realizadas por los clientes, ignorando la sucursal en la que se encuentra asociado.

A continuación se muestra el modelo entidad relación de la base de datos:

Figura 1. Diagrama entidad relación



Para lograr un trasformación adecuada del modelo entidad relación se deben tener en cuenta los siguientes pasos:
Identificar las tablas dependientes, para lograr su unión y determinar en qué orden se realizara. En el caso de estudio sería de la siguiente manera:

Tabla 4. Tablas dependientes

Tabla	Tablas dependientes
Préstamo	Pago_cuota
Tipo_pres	préstamo
Tipo_movi	movimiento
Cuenta	Tipo_pres y tipo_movi
Cliente	Tarjeta y Cuenta

El cuadro anterior da a conocer que todas las tablas serán integradas dentro del cliente que sería el valor principal, del cual se desea conocer las transacciones que realiza. Pero para manejar una mejor consulta de los datos se van a crear dos tablas: cliente y cuenta.

Integrar las tablas dependientes dentro de las principales. Para lo cual se debe conocer las llaves primarias, que se convertirán en las "row key". Además la tabla que se une se convierte en una familia columna de la principal. Cuando se integra una tabla con dos o más familia-columna se convierte en una súper-columna de la tabla principal.

A continuación se realizara el proceso para cada tabla del caso de estudio:

En la siguiente tabla se muestra la unión entre la tabla préstamo y la tabla pago_cuota, donde se crean dos familias: información y pago_cuota con sus respectivas columnas.

Tabla 5. Transformación tabla préstamo

Tabla	Row (key)	Familia	Columnas
préstamo	Cod_pres	información	Valor, fecha, año, cuotas
		Pago_cuota	Id_cuota valor_cuota, valor_interes

En la tabla N°6 se observa la unión entre la tabla tipo_pres y la tabla préstamo, donde se crean dos familias: información y préstamo. Como la tabla préstamo tenía dos familias se crea una súper-columna con nombre presta.

Tabla 6. Transformación tabla tipo-préstamo

Tabla	Row (key)	Familia	Súper-columna	Columnas
Tipo_pres	Cod_tpres	información		nombre
		préstamo	Presta	Valor, fecha, año, cuotas, Id_cuota valor_cuota, valor interés

En la siguiente tabla se muestra la unión entre las tablas tipo_movi y la tabla movimiento. En esta se crean dos familias: información y movimiento con sus respectivas columnas.

Tabla 7. Transformación tabla tipo-movimiento

Tabla	Row (key)	Familia	Columnas
Tipo_movi	Tipo_mov	información	nombre
		movimiento	Cod_mov, valor y fecha

En la tabla N°8 se realiza la unión entre las tablas cuenta, tipo_pres y tipo_mov, allí se crean tres familias: información, préstamo y movimiento. Como las tablas

tipo_pres y tipo_mo tenían cada una más de una familia, se crean dos súper-columnas con nombre tipo_pres y tipo_mov. Esta tabla se crea dentro del ambiente Big Data.

Tabla 8. Transformación tabla cuenta

Tabla	Row (key)	Familia	Súper-columna	Columnas
Cuenta	Cod_cuenta	información		Saldo
		préstamo	Tipo_pres	Nombre, valor, fecha, año, cuotas id_cuota valor_cuota, valor_interes.
		movimiento	Tipo_mov	nombre, código, valor y fecha

En la tabla N°9 se realiza la unión entre las tablas cliente, cuenta y tarjeta, donde se crean tres familias: información, cuenta y tarjeta. Esta es la segunda tabla que se crea dentro del motor NoSQL HBase.

Tabla 9. Transformación tabla cliente

Tabla	Row (key)	Familia	Columnas
Cliente	Cedula	Información	Nombre, apellido, teléfono, fecha_naci
		Cuenta	Cod_cuenta
		Tarjeta	Id_tarjeta, Cupo_inicial, saldo, fecha_entrega, fecha_cancelacion

5. CONCLUSIONES

La estructura de un ambiente Big Data ayuda a mejorar la manipulación de los datos, optimizando la gestión de la información respecto a tiempo y costo, logrando obtener mejores resultados en las estadísticas para una buena toma de decisiones.

La creación de un ambiente Big Data se debe realizar dentro de un cluster, el cual permita integrar todas las aplicaciones que se van a utilizar, como en este caso Hadoop, en el cual se almacena la información y las aplicaciones corren dentro del mismo nodo.

La trasformación de un modelo relacional a un modelo basado en columnas puede ser enfocada en diferentes caminos, dependiendo del punto de vista que se esté tomando o de los resultados que se desean saber durante los procesos de consulta.

6. AGRADECIMIENTOS

Agradecemos en primer lugar a Dios, por permitirnos llegar a esta meta por la que se ha trabajado tanto, a nuestras familias por todo el apoyo recibido, a nuestros amigos y compañeros de universidad y a todos los maestros que tuvimos durante la carrera, agradecemos a la profesora Claudia Milena Rodríguez Álvarez quien nos guio en gran parte del proyecto y al profesor Yasser de Jesús Muriel Perea, quien en esta última fase acepto asesorarnos para la finalización del proyecto.

7. BIBLIOGRAFIA

[1] (2013) Gartner. [Online] Available: http://www.gartner.com/it-glossary/big-data/

[2] (2012) Forrester [online] Available: http://blogs.forrester.com/mike_gualtieri/12-12-05-the_pragmatic_definition_of_big_data

[3] (2012) Routledge. [online] Available: http://www.tandfonline.com /doi/pdf/10.1080/1369118X.2012.678878

[4] (2012) University of Amsterdam [online] Available: http://bigdatawg.nist.gov/_uploadfiles/M0055_v1_7606723276.pdf

[5] (2012) SG [online] Available: http://sg.com. mx/content/view/966

[6] Nasholm Petter. Extraer datos de bases de datos NoSQL. 1 Ed. Gotemburgo 2012

[7] (2012) SG [online] Available: http://sg.com .mx/content/view/966

[8] (2010)Ayende@Rahien [online] Available: http://ayende.com/blog/4500/that-no-sql-thing-column-family-databases